Simulating Traffic with Queueing Models

Submission date: July 31, 2003
Word count: 5362 + (7 figures) * 250 words/figure = 7112

Nils Eissfeldt*, Center for Applied Computer Science (ZAIK), University of Cologne, Weyertal 80, 50931 Köln, Germany; Telephone: +49 (0)221 / 470-6026; fax: +49 (0)221 / 470-5160; e-mail: eissfeldt@zpr.uni-koeln.de
Jürgen Gräfe, Center for Applied Computer Science (ZAIK), University of Cologne, Weyertal 80, 50931 Köln, Germany; Telephone: +49 (0)221 / 470-6010; fax: +49 (0)221 / 470-5160; e-mail: graefe@zpr.uni-koeln.de
Peter Wagner, German Aerospace Center (DLR), Institute of Transport Research, Rutherfordstrasse 2, 12489 Berlin, Germany; Telephone: +49 (0)30 / 67055-237; fax: +49 (0)30 / 67055-202; e-mail: peter.wagner@dlr.de

* Corresponding author

ABSTRACT

Queueing models for traffic simulations are interesting models for applications. They can be used to simulate up to 10^8 cars/s on completely standard hard-ware. Unfortunately, uptonow, there was a catch: they sometimes show weird results, most notably jams that do not run backward. This work shows how queueing models can be made as realistic as most car-following models. Two ingredients are needed: the first is the proper translation of the macroscopic description (flows) into the microscopic headways used in the queueing model, the second is the correct modeling of the interaction between the queues that model the links in the network.

INTRODUCTION

Queueing models for microscopic traffic flow simulation have been used in a number of different groups worldwide (1-6), see (7) for additional references, including our own (8,9). The idea behind them is very simple: any link of a road network can be regarded as a queue of cars. If the link gets very long, it can be divided into several queues (which does not change the idea), but at first a link is identified with a queue. Upon entering the link, any car gets assigned a proposed leaving time, when this time has been elapsed, it can be put into the next link of its path - provided, some capacity constraint is obeyed at exiting and there is enough space on the destination link.

These kind of models play an important role in several fields of modelling traffic flow systems, espacially in the context of dynamic traffic assignment (DTA) and ITS applications (4). Their importance is due to the fact that they usually are computationally very efficient without neglecting important dynamic effects of traffic flow. E.g., with respect to the calculation of the user equilibrium for a given travel demand and traffic network iterative approaches are widely used. This assignment procedure can be devided into two main stages, namely the modelling of the route--choice behaviour and the dynamic network loading. For the latter purpose any traffic flow model can be used in principle if it is able to reproduce travel times sufficiently detailed. However, since the two stages of DTA have to be repeated several times (usually up to 10-40 times), computational very fast models are necessary in order to allow for extensive scenario calulations and on-line applications.

Queueing models for instance posses an appropriate balance between computational performance and proper modelling of traffic flow with respect to the network loading computation for huge networks. In (9), such a queueing model has been compared to a cellular automaton (10) and to a state-continuous, but time-discrete microscopic traffic flow model (11), with excellent results. However, when using the queue model to compute emissions (12), some strange results were encountered.

The basic point is that jams in this model do not run backward but stick at the position where they were initiated. We have the suspicion that this happens with almost all implementations of queueing models in use for the purpose of traffic simulations, except (2,4). This will be further discussed in the following section.

It turns out that the tests used in (9) succeeded because a certain, very special situation was utilized for the comparison: a two-lane road with a one lane bottleneck at the end. Which is the queueing scenario per se. Since the results of this comparison where so excellent, nobody realized that it needs more to built a successful queueing model.

The paper is structured as follows. In the next section it will be discussed in short what is missing in most queueing approaches. Due to the vast number of different approaches we can only focus on special aspects which seem important to us. Subsequently, we present our queueing approach which is able to fix the mentioned problems, in particular the modelling of backward moving jams. Before concluding, two benchmarks of the proposed model will be presented, testing the models' ability to reproduce measured travel times and measured loop detector data.

A SHORT REMARK ON QUEUEING APPROACHES

Microscopic modeling of individual car motion is less in the focus of approaches based on classical queueing theory than the connection of flow-density relations with relevant determinants of traffic flow. In terms of queueing theory each link or street segment is to be regarded as a service device operating at a certain service rate. Queues, i.e. congestion, occur whenever the current demand exceeds the capacity of the service. Thus, the resulting travel time on a link is the sum of waiting time plus service time. Depending on the distributions of arrival and service times assumed in such models certain relations for the travel time arise. E.g. assuming a M/M/1 queueing model (13) leads to a type of function Davidson proposed in (14), $t_{trav}(q) \propto 1 + (c\, q) / (q_{max} - q)$. Here q and q_{max} are the instant flow on the link and its maximum flow, respectively, c is a parameter which has to be estimated from field measurements.

In principle in these models the resulting travel time is always some kind of nonlinear function depending at most on the current state of the queue a car enters. This also holds if nonstationary queueing approaches (3) or state-dependent queueing models (7) are used. In all such models, spill-back with respect to a network is properly modeled. But regarding a sequence of queues that are congested (a jam) one does not observe backward propagation of the jam as one finds in reality (15) . Jams stick at the position where they were generated and queues solely disolve from the back until that position is reached. The lack in the concepts mentioned above is that there is no mechanism introduced to distinguish the two situations, namely the transmission of a car between two queues that are completely congested and the escaping of a car at the downstream front of a jam. Therefore we claim that a queueing model that properly models jamming has to posses service times that do also depend on the state of the queue in front. This point has also been worked out in the cell transmission model (CTM) by Daganzo (2) and the space-time queue concept by Mahut (4) nevertheless both following a different line of thought than we do. In these models the flow between two queues is dependent on the states of both queues, i.e. the sending and the receiving queue. However, in CTM flows are modeled as real valued numbers while we seek for a concept where individual drivers are represented, i.e. a discrete-flow model. In the model by Mahut a minimum time separation between two vehicles is introduced which is fixed by parameters of the flow-density relation. The temporal separation between two consecutive cars also plays a prominent role in the model we will propose in the following section. In contrast to the model by Mahut arguments are purley on the level of microscopic car motion.

IMPROVEMENT OF KNOWN APPROACHES

The following arguments are focused on the queueing model (FASTLANE) in (8,9) but from what was discussed above it is clear that they hold in general. This is, since almost all approaches result in a travel time function (composed by running time plus additional waiting times due to flow constraints at the exit of the link) which depend at most on the instant conditions of the link. In particular, FASTLANE fails, because of the implemented waiting time function:

Upon entering link i at time t_ν, a car ν gets assigned an estimated exit time $t_\nu^{exit} = t_\nu + t_{trav}$, where the travel time is given by $t_{trav}(n_i) = L_i\, v_i^{max}$. In general, this travel time depends on the number of cars already on the link n_i, however, in most cases it is choosen to be independent of n_i. The symbol L_i is the length of the link i and v_i^{max} is the speed limit on that link. If time t exceeds t_ν^{exit}, the car enters the next link, provided there is at least one site free, i.e. $n_{i+1} < N_{i+1}$, with N_{i+1} denoting the maximum number of cars on link $i+1$. An additional constraint is provided by the maximum flow Q_i which the cell can sustain. This maximum capacity, and, in general, any flow q has to be converted into the corresponding time headway between two cars. Since the flow is just $q = 1/\tau$ with τ being the temporal headway between two consecutive cars ν-1 and ν, this capacity constraint is easily written down as an additional waiting time τ_W.

$$\tau_W = 1/Q_i \tag{1}$$

where τ_W denotes the waiting time between the car ν-1 and the car ν, under consideration. Only after the time $t > t_\nu^{exit} + \tau_W$, the car is allowed to leave the link i. (The current implementation used in (9) to ensure the capacity constraint is different from what is proposed here, but that does not change the argument.). As mentioned already above, this simple recipe is not enough to make a valid traffic flow model. Surprisingly, even strongly nonlinear travel time functions such as $t_{trav}(n_i) \propto 1 + 1 / (1 - n_i / N_i)$ are not able to generate a backward moving jam. Note, that state-dependent queueing approaches finally result in such kind of functions for the travel time. So, as has been observed already in (9), the detailed form of the travel time function seems to be not very important. This can be understood, since the reason for the non-linearity of empirically observed travel time functions may be just the bottleneck at the end.

So what is the trick that transforms a queueing model into a useful traffic flow model? While the upstream jam front in the model behaves as expected, the downstream jam front is fixed. The reason for this is that the inflow into the queue that currently marks the downstream front of the jam is as big as the outflow from this queue, which is equivalent to the outflow from the jam. Therefore, the flaw in the reasonings above is to assume that the waiting time τ_W at the end of link i is independent of the number of cars in the next link $i+1$. This can be understood quite simply. Suppose the first car of an otherwise full link $i+1$ leaves this link at a certain time. In reality, the free site (hole in physical parliance) generated in this manner needs a certain time to travel backwards. Therefore, the first car in the upstream link has to wait until the hole has travelled to the end of the link. This needs a time that is proportional to the number of cars already on that link, therefore it is straightforward to assume in dense traffic:

$$\tau_W^{(i)} \propto n_{i+1} \tag{2}$$

Note, that for a homogeneous fleet of cars with the length l the relation $l\, N_{i+1} = L_{i+1}$ holds. Generalizing this idea, five parameters have to be introduced that determine the behaviour of the waiting time as function of the states of the two cells i and $i+1$. To distinguish between free flow (f) and congestion (j) a parameter n_{jam} is introduced and cell i is called to be jammed if $n_i \geq n_{jam}$. The four possible combinations of states are characterized by different waiting times which are named quite tellingly τ_{ff}, τ_{fj}, τ_{jf}, τ_{jj}, respectively. Therefore, the most general version of the queueing model reads:

$$\tau_W^{(i)} = \begin{cases} \tau_{ff} & & n_i < n_{jam} \quad and \quad n_{i+1} < n_{jam} \\ \tau_{fj} & if & n_i < n_{jam} \quad and \quad n_{i+1} \geq n_{jam} \\ \tau_{jf} & & n_i \geq n_{jam} \quad and \quad n_{i+1} < n_{jam} \\ f(\tau_{jj}, n_{i+1}) & & n_i \geq n_{jam} \quad and \quad n_{i+1} \geq n_{jam} \end{cases} \tag{3}$$

In principle, $f(\tau_{jj}, n_{i+1})$ reflects (2) and is specified by a linear realtionship here,

$$f(\tau_{jj}, n_{i+1}) = m n_{i+1} + b, \tag{4}$$

where m, b are chosen such that $f(\tau_{jj}, n_{i+1})$ for $\tau_{jf} = \tau_{jj}$,

$$m = \frac{N_{i+1} - 1}{N_{i+1} - n_{jam}} \tau_{jj} \qquad b = \tau_{jj} - n_{jam} m. \tag{5}$$

If not stated otherwise we will use $\tau_{jj} = \tau_{jf}$ in the following (see also below). Therefore f is a continuous function at $n = n_{jam}$. When a car ν-1 occupies the last place of a cell $i+1$, the corresponding waiting time for car ν at i is high enough to allow the cell in front to flush completely if it is the downstream front of a jam.

Of course, $\tau_{fj} = \tau_{ff}$ in most cases. In principle, the function τ_W introduced above can be made more complicated by taking care of the history of the system. E.g., at the end of a large jam the jam occupies only part of a queue, therefore for this case the relation $\tau_{fj} = \tau_{ff}$ applies. If the inflow into this link is smaller than its outflow, this is no longer correct, because in this case the jam-front may finally travel backward within one queue. However, this happens rarely. So, to keep this model simple, this effect will be ignored. Note, that it may play a role if the links in use becomes fairly long. For those long links, it may even happen that a microscopic simulation would yield a substructure of the link, e.g., with a jam in the middle. This is not captured by this model, to do it, the concept of moving waiting queues has to be introduced. Macroscopically, this has been done in the models invented in (16), it would be an interesting task to transform that into the microscopic approach suggested here. However, link-lengthes of several 100 m are used in other macroscopic models as well and can be assumed to be fairly uncritical.

Furthermore, when testing the model with real data, it seems for most purposes sufficient to work with just two parameters, τ_{ff} and τ_{jj}. Nevertheless, it is hypothesized that a queueing model with this five parameters can be used to mimic any of the microscopic simulation models currently under discussion, with only slight adaptations of the parameters. It can not, however, describe the hypothetical state that is known as synchronized flow, and therefore none of the models that claim to model synchronized flow (15,17).

Note, that similar ideas, which however still use a macroscopic framework (so called Markovian traffic flow models), have been put forward by a number of other researchers, cf. (18) and references therein; as mentioned before the cell transmission model (2) and the space-time queue concept (4), although derived through a completely different line of thought, uses similar ideas. Furthermore, some of the ideas here can also be found in (19).

A couple of simulations have been run with a queueing model implementing these ideas. To do so a one-lane-loop, i.e., periodic boundary conditions, of fixed length was used in a first step. It was divided into a sequence of several queues with equal length 100 m. Therefore each queue has the same storage capacity N_i. The queues are realised with FIFO queueing discipline. After system's initialisation, the state of each queue is updated in discrete time steps using a parallel update scheme. In a first step it is determined for each queue if there is a car allowed to leave, i.e., if

$$t \geq \max\left\{t_\nu^{exit}, t_{\nu-1}^{exit} + \tau_W(\nu-1, \nu)\right\} \quad and \quad n_{i+1} < N_{i+1}. \tag{6}$$

holds. If so, the queue gets assigned a new $\tau_W(\nu, \nu+1)$ according to (3). In a second step all cars that are allowed to move are put to the next queue on their path. Note that the update scheme can easily be transformed into an event--driven scheme improving computational efficiency.

In Figure 1 and 2 simulation results are shown for a loop which was initialized with two separated jams. It can be seen that the model is able to reproduce their stable movement against driving direction. Moreover, in the corresponding flow-density relation (fundamental diagram) one finds that there exist a capacity drop, i.e., the flow out of jam is noticably smaller than the maximum flow possible. This is due to the choice $\tau_{ff} < \tau_{jf}$, which compares to the slow-to-start behaviour known from microscopic car--following models (11,20).

Indeed, further investigations of the model show that it needs the condition $\tau_{ff} < \tau_{jf}$ in order to have stable moving jams. This can be seen even more vividly if one determines the critical density ρ^* above which jams are stable in the model depending on the parameters of the model.

Let τ_{esc} be the time that a jammed queue needs to flush completely if it marks the downstream end of a jam. Since jams in the model are compact, i.e., $n_i \approx N_i$ for a queue inside a jam, this is just the sum of the individual waiting times τ_ν assigned to each car ν,

$$\tau_{esc} = \sum_{\nu=1}^{N} \tau_\nu \qquad \text{with} \qquad \tau_\nu \in \tau_W \tag{7}$$

Given one jam in a loop with global density ρ, in the stationary state the number of cars inside the jam is then given through

$$N_j = L_S \left(1 + \frac{L}{Nh_{eq}}\right)\rho - \frac{L_S}{h_{eq}}, \tag{8}$$

where L_S, L are the system length, length of a cell, respectively, and

$$h_{eq} = \frac{v_{\max}}{N}\tau_{esc}. \tag{9}$$

The critical density ρ^* is then just a function of the model's parameters

$$\rho^* = \frac{1}{h_{eq}\left(1 + L/\left(Nh_{eq}\right)\right)}. \tag{10}$$

So, for small τ_{esc} , $\rho^* \rightarrow \rho_{max}$, i.e., there are no stable jams in the model. ρ^* can easily be determined by simulation. In Figure 3 this was done variing τ_{ff} and τ_{jf} and using (10) to calculate τ_{esc} . It can be seen, that there are two regimes, and only for $\tau_{ff} \leq \tau_{jf}$ there exist stable jams.

Moreover, this result can explain, why not all queueing approaches display stable backward running jams. As already mentioned above, it is important to account for the situation in front of the jam to determine additional waiting times instead of solely the state of the queue itself. It even turns out that the known capacity drop in the outflow region of jams has to be respected on a microscopic level of modelling traffic flows. On the other hand, a proper model of jams has to take into account the limited velocity a perturbation takes to travel backward (e.g. this is also done by the models in (2,4). If this requisite is not met a quasi simultaneous dissolution of a jam takes place. This is the case, where $\tau_{esc} \rightarrow 0$.

Note, that the model we presented is a deterministic one. Therefore, there is no mechanism to generate a jam by intrinsic noise. Anyway, in a complex network with on- and off--ramps and signalized intersections there exist plenty of inhomogeneities which can lead to queue formation (cf. next section).

BENCHMARKING THE QUEUE MODEL

In the following we will present two benchmarks of the model using real world data. Doing so, two questions will be answered: (i) how successful can such a model describe reality, (ii) how robust is it. In more detail we want to show, that the queueing approach presented is able to capture dynamic situations as the building of a queue on a highway section between two entrances and that its properties are quite robust against the choice of parameters.

Choice of parameters

The first data-set that has been used for calibration and validation has been recorded by Daganzo and co-workers several years ago (21,22). Along a one-lane road where overtaking was almost impossible, eight observers were positioned, each of them equipped with a lap-top to record the times a certain car passes him/her. One lead car that drove several times along this road provided the initial car number, therefore traffic flow is recorded by $N_i(t)$-curves where $N(t)$ is the number of cars that passed a certain position (observer i) up to a certain time t. Typical recording errors are of the order of 5 cars out of 600 by which the counts of different observers differed. The data set consists of two parts recorded on two different days, one day has been used for calibrating, the other just for validating the model. For further details, refer to (21,23).

The model used here is a slightly modified (more general) version of the queueing model used in (23). The $N_1(t)$-curve at the left boundary (entry) was used to feed cars into the system. The right boundary (exit) of the system was also fixed using the corresponding cumulative counts. Cars can immediately leave the system if they arrive in time or delayed otherwise they have to wait at the exit until their exit time given through $N_8(t)$ has been reached. At the intermediate six observer positions the difference between the travel time from the simulation and the measurements is calculated for each car giving the error done by simulation. In order to determine the optimal parameters for the queueing model we used the downhill simplex method (DMS) described in (24). As error function we defined the average error recorded at each intermediate observation spot. When fitting all parameters, the following results were obtained: v_{max} = 21.7 m/s, τ_{ff} = 1.38 s, τ_{jf} = 0.8 s, τ_{fj} = 0.78 s, τ_{jj} = 0.57 s, with an error of 16.5%. This error compares to the error detailed microscopic car-following models do in predicting travel times on the same input data (23). The length of a queue was fixed to L = 100 m (N = 14, n_{jam} = 5). Using the three-τ ($\tau_{fj} = \tau_{ff}$) or the two-τ ($\tau_{fj} = \tau_{ff}$, $\tau_{jf} = \tau_{jj}$) version of the model gives the same result with respect to the error function.

For applications of such a queueing model it seems interesting to make the queues as big as possbile. Therefore, another test with this data set has been performed, where the optimal set of parameters as function of the queue length L has been determined. Note, that by too long segments an additional error is added, since the observers get displaced with respect of the endpoints of the segments. Nevertheless, even for larger queue-lengths the error remains tolerable, see Figure 4, enabling the use of this model in serious applications.

Dynamic queueing

The second data-set was measured at Gardiner Expressway, a freeway which is located in metropolitan Toronto, Canada. After an on-ramp there is a measurement station (Station1) followed by a three-lane segment of almost 2.5 km before cars can leave the freeway. In between there are three more stations where measurements are taken. If one takes Station 1 as reference point, Station 2, 3 and 4 lie at positions 780 m, 1360 m and 1850 m, respectively. The inductive loop detectors record vehicle counts, occupancies and time mean speeds in each lane at 20-seconds intervals.

The dynamic situation for the day we chose is quite interesting. More than one kilometer downstream of the merge at Station1 one finds a bottleneck for a certain period of the day. This can easily be seen if one has a lock at the mean speeds at the different measurement stations (cf. Figure 5). For further explanation of the situation see (25). Note, that the traffic breakdown is almost not visible at the right boundary at Station4.

In order to use the data as a benchmark for our model the $N_i(t)$-curves correponding to the four stations would be appropriate (cf. subsection above). However, when extracting the cumulative counts from the measured vehicle counts, one finds inconsistencies, namely that there are periods in which the maximum storage capacity of the segment is terribly exceeded. These inconsistencies are due to detector errors and we could not find a way to fix them by rescaling the curves. Therefore, we only used the $N_1(t)$-curve at Station1 to feed our system. Additionally, we mapped the inflow to a one-lane situation by dividing the vehicle counts through the number of lanes. Geometry was fixed in terms of cells with length L = 100 m. Due to the lack of consistent cumulative curves at the other stations we used the measured velocities instead. At the exit, two additional cells were used (i.e. lying outside the system) which are controlled by the velocity measured at either Station3 or Station4, i.e. if a car enters at a specific time step, its free flow velocity is set to the corresponding value. At the intermediate points the measured values of the velocity are used to define an error function by comparing them with the simulated velocities. Again we used the DSM to determine the optimal paremeters of the queueing model.

In a first scenario we only used the first three stations, i.e. the right boundary was controlled by Station3 and the error was calculated at Station2 and Station3. Since it is known, that DSM has difficulties fitting integer valued parameters, N and n_{jam} were fixed to 14 and 2, respectively. For the other parameters the optimization procedure yielded v_{max} = 26.9 m/s, τ_{ff} = 1.55 s, τ_{jf} = 1.67 s, τ_{fj} = 2.52 s, τ_{jj} = 2.8 s with an average error of 10 m/s and 4.9 m/s at Station2 and Station3, respectively. The resulting velocity curves are plotted in Figure 6. Both breakdowns that occurred during the day are reproduced very well. Due to the deterministic nature of the model the curves do not fluctuate as strong as the measured values. Note, that the parameters yield $\tau_{ff} \leq \tau_{jf}$ necessary for stable jamming in the model and $\tau_{ff} \approx \tau_{fj}$ and $\tau_{jf} \approx \tau_{jj}$ (cf. discussion of the model above).

In a second scenario we used all four stations. From the study (25) it is known that a dynamic bottleneck is observed around 1 km downstream the entry at Station1. It can be suspected that this is due to a change in conditions of the street (in the original geometry there is a curve). We therefore splittet the geometry into two segments. The cut is located at 1 km, i.e. 200 m after Station2. The parameters N and n_{jam} were fixed for both segments as given above. Appart from this we assumed distinct parameter sets for both segments. The optimization procedure yielded the following results: a) For the section from 0 to 1000 m: v_{max} = 27 m/s, τ_{ff} = 2.85 s, τ_{jf} = 3.17 s, τ_{fj} = 2.06 s, τ_{jj} = 3.05 s. b) For the section from 1000 m to 1900 m: v_{max} = 26.4 m/s, τ_{ff} = 1.8 s, τ_{jf} = 2.66 s, τ_{fj} = 2.36 s, τ_{jj} = 1.88 s. The average error at Station2, Station3 and Station4 was 17.9 m/s, 11.7 m/s and 6.9 m/s , respectively, the resulting velocity curves are plottet in Figure 7. Note, that the higher average error results from the fact, that the first breakdown is not really reproduced by the model. Recall, that this breakdown is almost not visible in the velocity curve which has been used to control the system's exit. It demonstrates, what was said before, namely that the queueing model is much more sensible to cumulative counts than to velocities. This has been tested with some fictitious cumulative curves which reflect a dynamic situation similar to the one used here. The agreement that we could achieve there was considerably higher. Nevertheless, also in this scenario the second breakdown is reproduced with good quality. Moreover, τ_{ff} for section 1 is bigger than that of section 2 which reflects the fact that somewhere between Station2 and Station3 a bottleneck is present.

The results show, that the queueing model is quite able to reproduce a dynamic situation that was taken from real-world data although it is based on a deterministic picture of jamming.

SUMMARY

It has been shown, that queueing models can be used for doing microscopic and realistic traffic simulations. This work has made explicit a slight generalization which includes the interaction to the link downstream. Additionally, by proposing the appropriate translation from macroscopic flows into the microscopic headways used in queueing models, these models may be used in several applications. The results also demonstrate that they cannot be used for really high-level considerations like the weaving flows at intersections, but are very much appropriate for large-scale applications like stochastic dynamic user equilibria or the reconstruction of the system state with the help of measured data (26). For those scientists who believe in the spontaneous break-down of traffic flow, the model can easily be extented by imposing some stochastic delays on the waiting times. Even not demonstrated here, we implemented this probabilistic scheme using Erlang distributions and indeed the model is then able to generate stable jams out of nowhere. Note, that the comparison with real data that has been pursued in this work did not need that. It is amazing, that the little microscopic knowledge explicit in the queueing models is sufficient to model the macroscopic (the jams) and the large-scale features (like the travel times) of traffic.

ACKNOWLEDGEMENTS

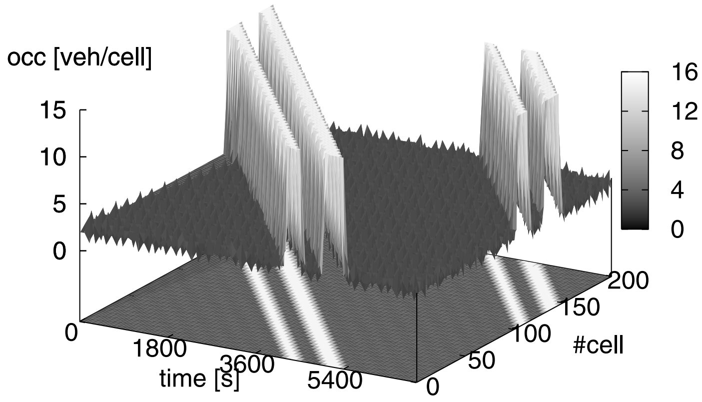
PW learned very much about the matters discussed here during a stay at the ITS in Berkeley. The discussions with Carlos Daganzo are gratefully acknowledged. Parts of the ideas put forward here are currently introduced into a project funded by the German Ministry for Education and Science. The authors also thank R. Bertini for the possibility to use the measurements from Gardiner Expressway.

REFERENCES

1. Akamatsu, T. “An efficient algorithm for dynamic traffic assignment with queues”. *Transportation Science*, 35(4):389-404, 2001.
2. Daganzo, C. “The cell transmission model : A dynamic representation of highway traffic consistent with the hydrodynamic theory”. *Transportation Research B*, 28(4):269-287, 1994.
3. Heidemann, D. “A queueing theory model of nonstationary traffic flow”. *Transportation Science*, 35(4):405-412, 2001.
4. Mahut, M., Florian, M. and Tremblay, N. “Space-time queues and dynamic traffic assignment: A model, algorithm and applications”, TRB 2003 Annual Meeting CD-ROM, 2003.
5. TRANSIMS, TRansportation ANalysis and SIMulation System, see http://transims.tsasa.lanl.gov
6. Vandaele, N., Woensel, T.V. and Verbruggen, A. “A queueing based traffic flow model”. *Transportation Research D*, 5(2):121-135, 2000.
7. Jain, R. and Smith, J. “Modeling vehicular traffic flow using M/G/C/C state dependent queueing models”. *Transportation Science*, 31(4):324-336, 1997.
8. Gawron, C. “An iterative algorithm to determine the dynamic user equilibrium in a traffic simulation model”. *International Journal of Modern Physics C*, 9(3):393-407, 1998.
9. Gawron, C. “Simulation-based traffic assignment”. PhD thesis, ZAIK, University of Cologne, available at http://www.zaik.de/AFS/theses.html, 1998.
10. Nagel, K. and Schreckenberg, M. “A cellular automaton model for freeway traffic”. *Journal de Physique*, 2:2221-2229, 1992.
11. Krauß, S. “Towards a unified view of microscopic traffic flow theories”. In M. Papageorgiou and A. Pouliezos, editors, *IFAC Transportation Systems 1997*, 2:941-946, 1997.
12. Eissfeldt, N. and Schrader, R. “Calculation of street traffic emissions with a queueing model”. *Journal of Computational Technologies*, 7:5-15, 2002.
13. Gross, D. and Harris, C. “Fundamentals of Queueing Theory”. John Wiley & Sons, New York, 3rd edition, 1998.
14. Davidson, K. “A flow-travel time relationship for use in transportation planing”. In *Proceedings of the Australian Road Research Board*, 3:183-194, 1966.
15. Kerner, B. and Rehborn, H. “Experimental properties of complexity in traffic flow”.*Phys. Rev. E*, 53:R4275, 1996.
16. Kerner, B., Rehborn, H., Aleksic, M. and Haug, A. „Verfolgung und Vorhersage von Verkehrsstörungen auf Autobahnen mit ‚ASDA’ und ‚FOTO’ im online-Betrieb in der Verkehrsrechnerleitzentrale Rüsselsheim“. *Strassenverkehrstechnik*, 10, 2000.
17. Kerner, B. “The physics of traffic”. *Physics World*, 12:25-30, 1999.
18. Lan, C.-J. and Davies, G.A. “Empirical assessment of a Markovian traffic flow model”. *Transportation Research Records*, 1591:31-37, 1997.
19. Newell, G.F. “A moving bottleneck”. *Transportation Research B*, 27:281, 1993.
20. Barlovic, R., Santen, L., Schadschneider, A. and Schreckenberg, M. “Metastable states in cellular automata for traffic flow”. *European Physical Journal*, 5:793, 1998.
21. Smilowitz, K.R. and Daganzo, C.F. “Experimental verification of time-dependent accumulation predictions in congested traffic”. *Transportation Research Records*, 1710:85-95, 2000.
22. See http://www.ce.berkeley.edu/~daganzo
23. Brockfeld, E., Kühne, R.D., Wagner, P. and Skabardonis, A. “Towards a benchmarking of microscopic traffic flow models”. TRB 2003 Annual Meeting CD-ROM, to be published in TRR 2003, 2003.
24. Press, W.H., Teukolsky, S.A., Vetterling, W.T. and Flannery, B.P. “Numerical Recipes in C”, chapter 10, Cambridge University Press and references therein, 2002.
25. Bertini, R.L. and Cassidy, M.J. “Some observed queue discharge features at a freeway bottleneck downstream of a merge”. *Transportation Research A*, 36(8):683-697, 2002.
26. von der Ruhren, S., Beckmann, K.J., Eissfeldt, N., Gräfe, J., Mühlhans, H. and Wagner, P. „Simulationsbasierte Kurzfristprognose von Verkehrszuständen im Rahmen von stadtinfoköln - Methodik, Umsetzung, Erfahrungen“, to be presented at 19. Verkehrswissenschaftliche Tage, Dresden, 2003.

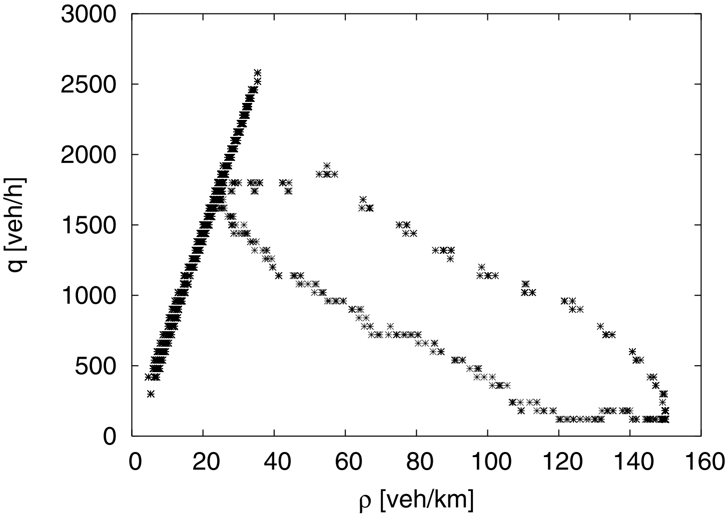
LIST OF FIGURES

FIGURE 1



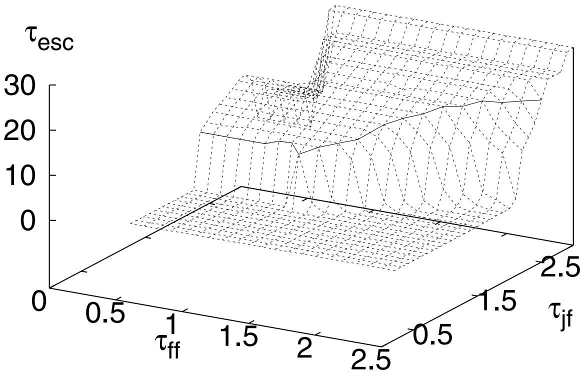
Space time plot of the number of cars within each cell. The parameters used in this simulation are: cell capacity $N = 14$ (i.e. maximum numbers of car per cell); minimum travel time $t_{travel} = 5$ s, independent of the number of cars in the cell; $n_{jam} = 5$; waiting times between cars: $\tau_{ff} = \tau_{fj} = 1.2$ s, $\tau_{jf} = \tau_{jj} = 2$ s. The system has been initialized with two separated jams, and it was simulated with periodic boundary conditions. The update scheme was parallel, with an update time of 0.1 s.

FIGURE 2



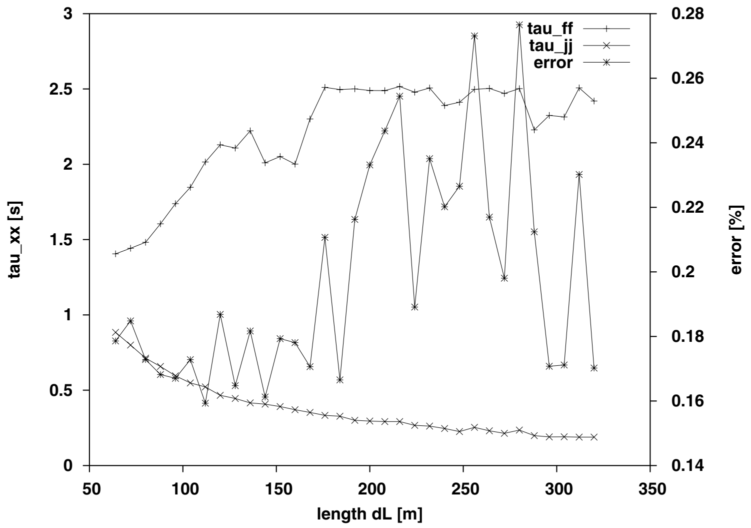
Fundamental diagram corresponding to the system presented in Figure 1. One can clearly see that the model is able to reproduce stable wide moving jams and the flow-density relation shows a capacity drop.

FIGURE 3



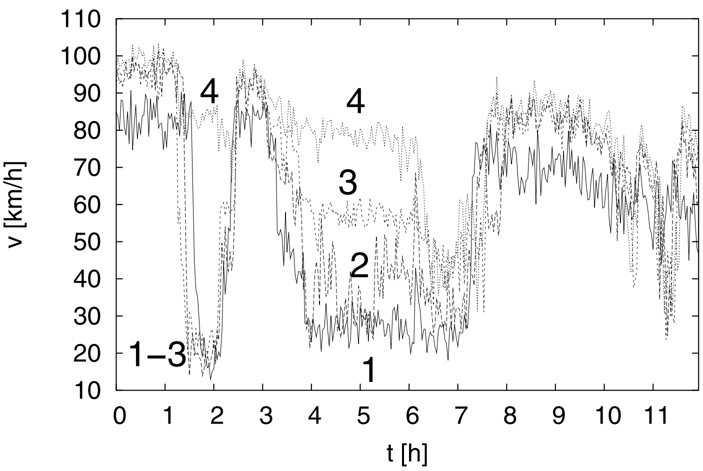
Escape time from jams versus τ_{ff} and τ_{jf} . The free flow speed was v_{max} = 20 m/s and each queue had length L = 100 m, i.e. N = 14. The model shows two distinct regimes with a sharp change in its behaviour. The transition region is marked by the contour line. Stable jams exist for $\tau_{jf} \geq \tau_{ff}$.

FIGURE 4



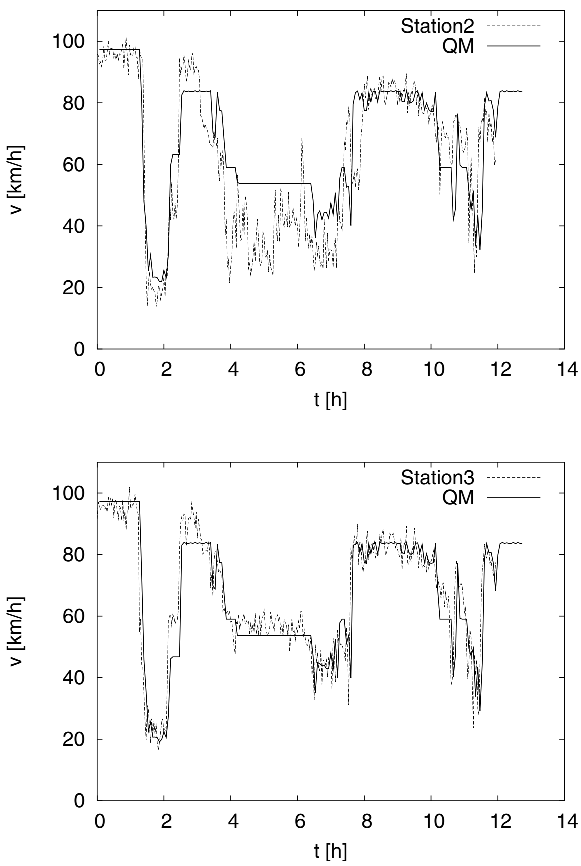
The modelling error, and the parameters respectively, plotted as a function of the length of the queue. For very short queues (not shown), and again for long ones, the error increases. Note, that any point in this curves is the result of an automatic optimisation process, which may or may not have succeeded. For very long queues, there is an additional error related to the fact, that the observer positions are more and more off the interfaces between the various queues.

FIGURE 5



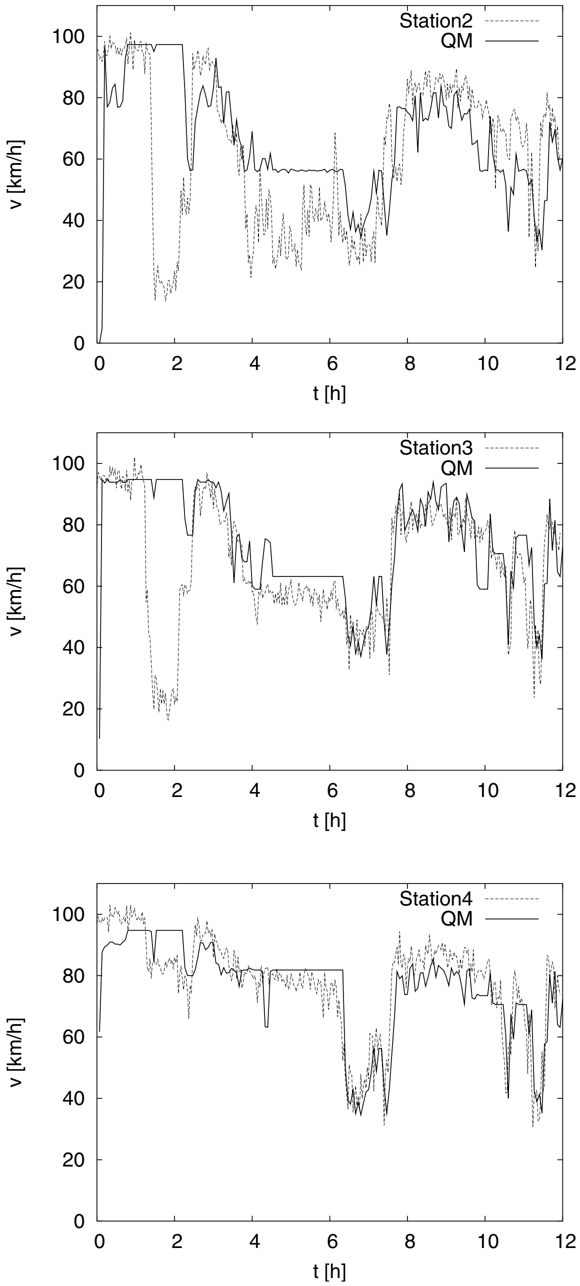
The dynamic situation which is used for the benchmark of the queueing model. Plottet is the velocity over time that was measured at the stations 1 to 4. One can clearly see that traffic breaks down in the middle of the segment for a certain period (Station2) while this breakdown is almost not visible at the exit of the segment (Station4).

FIGURE 6



Comparison between the velocities of the simulation and measurements at Station2 and Station3. All cells of the system are parametrized by the same set of parameters. The exit of the system was controlled by the velocities measured at Station3.

FIGURE 7



Comparison between the velocities of the simulation and measurements at Station 2 to 4. The system was devided into two segments at x = 1 km. For both segments a distinct parameter set has been used. The exit of the system was controlled by the velocities measured at Station4.